STOCHASTIC UNDERSTANDING MODELS GUIDED BY CONNECTIONIST DIALOGUE ACTS DETECTION

Emilio Sanchis, María José Castro and David Vilar

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València
Camí de Vera s/n, 46022 València, Spain
{esanchis,mcastro,dvilar}@dsic.upv.es

ABSTRACT

We study the use of specific stochastic models for the understanding process in a spoken dialogue system. A previous classification of the user turns in terms of dialogue acts is accomplished by connectionist models to guide the understanding process. Some specific issues are explored, like the multiclass classification problem, the smoothing of models, and the generation of the frames which constitute the input of the dialogue manager. Some experiments using the correct transcription of the user turns and the output of the speech recognizer are presented.

1. INTRODUCTION

Dialogue act (DA) classification is an important issue in the framework of the development of interactive speech systems such as dialogue or language translation systems. Many advantages can be achieved if we can detect and model the user and the machine sentences in terms of DAs. For example, stochastic models can be learnt to describe a dialogue behavior, to give predictions about the next user utterance, and so to focus the process of recognition or understanding, by means of specific rules or models [1, 2]. DA classification can also be useful in the domain of machine translation [3].

An important decision when a DA modelization is performed is the definition of the set of DAs [4]. If the number of DAs is small, each DA represents a general intention. If the number of labels is increased, each DA represents much more specific information, but many more training samples are needed. Our BASURDE speech dialogue system is a system for information retrieval by telephone for Spanish nation wide trains. We have defined a set of three-level DAs [5]. The first level tries to represent general actions of dialogues, such as Opening, Closing, Answer, ... The second level is related to the semantic of the task and the third level takes into account the values supplied in the utterances. With this type of DAs we can construct general models of the dialogue behavior, by using only the first level of DAs or more detailed models, by using more specific characterization of the utterances, that is, by using the second and third level.

In this work we present an approach to DA classification to help the understanding process of a dialogue system. Due to the inherent difficulty of the recognition and the understanding process of spontaneous speech in a mixed initiative dialogue system, the use of different analysis or classification of the user utterance should help obtaining better results. One of the specific characteristics, which makes the classification of a utterance in terms of DAs harder, is that an utterance can be composed by more than one DA. That is, if we try to adapt pattern classification techniques we encounter a problem of multiclass classification. We have explored the use of neural networks expecting that it capability of discriminative learning gives good results.

On the other hand, in this paper we also present the stochastic approach to the understanding process of the dialogue system. The process is based on the use of Hidden Markov Models (HMMs) to represent the available sequences of concepts (or semantic units), and the composition of this semantic units in terms of sequences of words [6–8]. The collaboration between the classification process and the understanding process is done by means of the definition of specific semantic models for each DA. That is, we first classify the user utterance as one or more DAs and then we apply the specific model to extract the semantic information, which is made by means of frames. Then the extraction of the frame (or frames) associated to one utterance is made in two phases. In the first one, a sentence (sequences of words) is transduced in terms of a sequence of semantic units, and its corresponding segmentation. In the second one, this intermediate representation is used to obtain the frames and attributes by using a set of few rules.

Thanks to the Spanish CICYT agency under contracts TIC2000-0664-C02-01 and TIC2002-04103-C03-03 for funding.

2. THE DIALOGUE TASK

The BASURDE task consists of information retrieval by telephone for Spanish nation-wide trains. Queries are restricted to timetables, prices and services for long distance trains. Several other dialogue projects [9, 10] selected similar tasks.

Four types of scenarios were defined (departure/arrival time for a one-way trip, departure/arrival time for a two-way trip, prices and services, and one free scenario). After that a total of 215 dialogues were acquired using the Wizard of Oz technique. From these dialogues, a total of 1 440 user turns (14 923 words with a lexicon of 637 words) were obtained. The average length of a user turn is 10.27 words. Figure 1 shows a fragment of a dialogue of the task.

2.1. Labeling the Turns of the Dialogues

The definition of DAs is an important issue because they represent the successive states of the dialogue. The main feature of the proposed labeling is the division of DAs into three levels [5]. The first level, called *speech act*, is general for all the possible tasks and it comprises the following labels: Opening, Closing, Question, Consult, Acceptance, Rejection, Confirmation, Answer, Undefined, Not Understood, Waiting. An example of the first level labeling of a fragment of a dialogue is shown in Figure 1.

The second and third level, called *frames* and *cases*, respectively, are specific to the task and give the semantic representation. The frames determine the type of communication of the user turn. Table 1 shows the 15 frame classes, along with their frequencies. Each frame has a set of attributes (cases) which have to be filled to make a query or which are filled by the retrieved data after the query. Examples of cases for this task are: Origin, Destination, Departure_time, Train_type, Price...

An example of the three-level labeling for some user turns is given in Figure 2. Note that each user turn can be labeled with more than one frame label (as in the second example of Figure 2), which allows a better specification of the meaning of the user turn, but it makes the classification and understanding processes harder (see Sections 3.1 and 4).

2.2. Lexicon of the User Turns

For classification and understanding purposes, we are concerned with the semantics of the words present in the user turn of a dialogue, but not with the morphological forms of the words themselves. Thus, in order to reduce the size of the input lexicon, we decided to use (general and task-specific) categories and lemmas. In this way, we reduced the size of the lexicon from 637 to 311 words.

Table 1. The 15 frame classes and their frequencies given as percentages of the total number of user turns in the overall corpus (1 440 user turns).

Frame class	%
Affirmation	26.75
Departure_time	18.27
New_data	13.16
Price	12.29
Closing	10.07
Return_departure_time	5.30
Rejection	4.34
Arrival_time	3.57
Train_type	3.37
Confirmation	1.73
Not_understood	0.63
Trip_length	0.24
Return_price	0.19
Return_train_type	0.05
Return_arrival_time	0.05

Table 2. Partition of the dataset for the experiment. For each type of experiment, the dataset was split into a training set (80% of the data) and a test set (20% of the data).

Experiment	Total	Uniclass (UC)	Multiclass (MC)
Training	1 071	692 (65%)	379 (35%)
Test	268	175 (65%)	93 (35%)

2.3. The BASURDE dataset

The dataset is composed of 1 338 user turns after discarding the sentences labeled with the less-frequent frame classes, which comprised the 10 first frame classes of Table 1. The dataset was randomly split (but we guarantee that each frame class is represented in the training and test set) so that about 80% of the user turns are used for training and the rest for testing. The partition of the data, along with the relative frequency of uniclass and multiclass samples, are shown in Table 2.

3. CONNECTIONIST DIALOGUE ACT DETECTION

Multilayer Perceptrons (MLPs) are the most common artificial neural networks used for classification. For this purpose, the number of output units is defined as the number of classes, $|\mathcal{C}|$, and the input layer must hold the input samples. Each unit in the (first) hidden layer forms a hyperplane in the pattern space; boundaries between classes can be approximated by hyperplanes. If a sigmoid activation function is used, MLPs can form smooth decision boundaries which are suitable to perform classification tasks [11].

For uniclass samples, the activation level of an output

Speaker		Sentence		Speech act
M	Bienvenido al sistema automático de información de trenes regionales y de largo recorrido, ¿qué desea?	*Welcome to the information system for nation-wide train, what information would you like?*	Opening
U	Quería saber los horarios del Euromed Barcelona-Valencia.	*I would like to know the timetables of the Euromed train from Barcelona to Valencia.*	Question
M	¿Quiere ir de Barcelona a Valencia?	*Do you wish to travel from Barcelona to Valencia?*	Confirmation
U	Sí.	*Yes.*	Affirmation
M	¿Para viajar hoy mismo?	*Do you wish to travel today?*	Confirmation
U	No, el próximo jueves.	*No, next Thursday.*	Answer
M	Le consulto los horarios de Barcelona a Valencia para el día 15 de julio. Un momento, por favor.	*I am looking timetables form Barcelona to Valencia for the 15th of July. One moment, please.*	Waiting

Fig. 1. A fragment of a labeled dialogue of the task. The first column indicates the speaker: Machine (M) or User turn (U); the second column the original sentence and its English translation; and the last column the first level label (*speech act*).

Original sentence	Quería saber los horarios del Euromed Barcelona–Valencia. *I would like to know the timetables of the Euromed train from Barcelona to Valencia.*
1st level (speech act)	Question
2nd level (frames)	Departure_time
3rd level (cases)	Departure_time (Origin: barcelona, Destination: valencia, Train_type: euromed)
Original sentence	Hola, buenos días. Me gustaría saber el precio y los horarios que hay para un billete de tren de Barcelona a La Coruña el 22 de diciembre, por favor. *Hello, good morning. I would like to know the price and timetables of a train from Barcelona to La Coruña for the 22nd of December, please.*
1st level (speech act)	Question
2nd level (frames)	Price, Departure_time
3rd level (cases)	Price (Origin: barcelona, Destination: la_coruña, Departure_time: 12/22/2003) Departure_time (Origin: barcelona, Destination: la_coruña, Departure_time: 12/22/2003)

Fig. 2. Example of the three-level labeling for two user turns. The Spanish original sentence and its English translation are given.

unit can be interpreted as an approximation of the a posteriori probability that the input sample belongs to the corresponding class [12]. Therefore, given an input sample $\mathbf{x}$, the trained MLP computes $g_k(\mathbf{x}, \omega)$ (the k-th output of the MLP with parameters ω given the input sample $\mathbf{x}$) which is an approximation of the a posteriori probability $\Pr(k|\mathbf{x})$. Thus, for MLP classifiers we can use the uniclass classification rule

$$k^\star(\mathbf{x}) = \operatorname*{argmax}_{k \in \mathcal{C}} \Pr(k|\mathbf{x}) \approx \operatorname*{argmax}_{k \in \mathcal{C}} g_k(\mathbf{x}, \omega)\,. \quad (1)$$

3.1. Uniclass and Multiclass User Turns

In contrast to the well-known uniclass classification problem, in some real-world learning tasks, a pattern can belong to more than one class from the set of classes $\mathcal{C}$. For example, in many important document classification tasks, documents may each be associated with multiple class labels. A similar example is found in our classification problem of DAs: a user turn can be labeled with more than one frame label[1] (as in the second example of Figure 2). In this case, the training set is composed of pairs of the form[2]

$$\{(\mathbf{x}_n, C_n)\}_{n=1}^{N}\,, \quad C_n \subseteq \mathcal{C}\,. \quad (2)$$

The multiclass classification problem is much harder to solve than the uniclass classification problem. In this work, we have treated each class as a separate binary classification problem (as in [14]), assigning a binary string of length $|\mathcal{C}|$ to each class $c \in \mathcal{C}$ or set of classes $C \subseteq \mathcal{C}$. During training, for a pattern from classes Price and Departure_time, for example, the desired outputs of these binary functions are specified by the corresponding units for those classes. With MLPs, these binary functions can be implemented by the $|\mathcal{C}|$ output units of a single network.

In this case, the multiclass classification rule is redefined as: an input sample $\mathbf{x}$ can be classified in the classes $K^\star(\mathbf{x})$ with a posteriori probability above a threshold $\mathcal{T}$

$$\begin{aligned} K^\star(\mathbf{x}) &= \{k \in \mathcal{C} \mid \Pr(k|\mathbf{x}) \geq \mathcal{T}\} \\ &\approx \{k \in \mathcal{C} \mid g_k(\mathbf{x}, \omega) \geq \mathcal{T}\}\,, \end{aligned} \quad (3)$$

being $g_k(\mathbf{x}, \omega)$ the k-th output of an MLP classifier with parameters ω given the input sample $\mathbf{x}$.

[1] In related works of DA classification [13], a hand-segmentation of the user turns was needed in order to have sentence-level units (utterances) which corresponded to a unique DA.

[2] The uniclass classification problem is a special case in which $|C_n| = 1$ for all samples.

3.2. Codification of the User Turns for the MLP

After the processes explained in section 2.2, we discarded those words with a frequency lower than five, obtaining a lexicon of 120 words. Note that sentences which contained those words are not eliminated from the corpus, only those words from the sentence are deleted. We think that for the task of DA detection the sequential structure of the sentence is not fundamental to classifying the type of frame.[3] For that reason, the words of the preprocessed sentence were all encoded with a local coding: a 120-dimensional bit-vector, one position for each word of the lexicon. When the word appears in the sentence, its corresponding unit is set to 1, otherwise, its unit is set to 0.

3.3. Training the MLP

With any neural network algorithm, several parameters must be chosen by the user. For the MLPs, we must select the network topology and their initialization, the training algorithm and their parameters and the stopping criteria [11, 12, 15]. Tests were conducted using different network topologies of increasing number of weights. In every case, a sigmoid activation function was used in all units. Experiments with the incremental version of the backpropagation algorithm, with and without momentum term, and the quickprop algorithm were performed. The influence of their parameters such as learning rate or momentum term was also studied. We selected all the parameters to optimize performance on a validation set: the training set is subdivided into a subtraining set and a validation set (20% of the training data was randomly selected for validation). While training on the subtraining set, we observed generalization performance on the validation set (measured as the mean square error) to determine the optimal setting of configuration (network topology and parameters of the learning algorithm) and the best point at which to stop training. Random presentation of the training samples was used in the training process. The threshold $\mathcal{T}$ of the multiclass classification rule (see equation (3)) is also learnt in the training process: we performed classification with the optimal configuration of MLP on the patterns of the validation set, proving several values of the thresholds and keeping the best one. Finally, we measure network performance on the test set for the best configuration and the learnt threshold.

3.4. Classification Performance of the MLP

We have considered a sample as correctly classified if the set of the original frame classes is detected. That is, if a user turn is labeled with two frame classes only and exactly those classes should be detected. The global classification rate was of 11.19%, 8.00% for the uniclass set and 17.20% for the multiclass samples. If we test the trained MLPs with the recognized test data obtained with our automatic speech recognition system (around 20% of word error rate), the classification error rate rises to a 48.13% (see Table 3).

Table 3. Frame type classification error rates of the user turns.

Experiment	Test	UC	MC
Transcribed data	11.19	8.00	17.20
Recognized data	48.13	50.86	43.10

Analyzing the output of the speech recognizer, we realized that many insertion errors of short and relevant words (e.g. "Sí" [*Yes*], "No") leads to a high classification error. These errors can be alleviated in some way in the understanding process (see Section 5).

4. STOCHASTIC UNDERSTANDING MODELS

Once we have classified the user utterances in one or more of the above defined frame classes, the next task in the dialogue system consists of extracting the relevant information in order to fill the cases associated which each user turn. To achieve this, we perform an additional step, finding an adequate segmentation of the user turn, according to the semantic function of each word or sequence of words (see Figure 3). The objective of this analysis phase is to find a correct segmentation according to this newly defined units. Having this segmentation, the filling of the corresponding cases can be accomplished with a set of simple rules. We defined a total of 53 semantic units [7], such as m_origen ("*departure_mark*"), clase_billete ("*ticket_class*") or fecha_actual ("*current_date*").

The segmentation is achieved using HMMs, where each state corresponds to a semantic unit. In order to reduce the size of the vocabulary and trying to avoid the problem of underestimation of parameters, we used the corpus obtained after the tokenization and lemmatization (see section 2.2).

We used a set of specific models for each of the 10 most frequent frame types defined in Table 1. The emission probabilities for each state in the HMMs will be shared between models, to avoid the problem of underestimation. The difference among the models will therefore lie in the transition probabilities between the different states conforming the model.

Here we must again face the problem of the multiclass user turns, as we have to combine the output of several HMMs, that is, we have to find an adequate combination of several segmentations, each from a different type of frame. In the training process we replicate the multiclass turns and use them to train each of the corresponding models. In the test phase, if the output of the classification provides more than one frame class, the corresponding HMMs are con-

[3] Nevertheless, the sequential structure of the sentence is essential in order to segment the user turn into slots to have a real understanding of it.

Original sentence			
Necesito saber los horarios de trenes de León a Córdoba para el tercer domingo de agosto.			
I need to know the timetable of the trains from León to Córdoba for the third Sunday of August.			
Segmentation			
necesito saber:	consulta	*I need to know:*	*question*
los horarios de trenes:	<hora_s>	*the timetable of the trains:*	*<time_d>*
de:	m_origen	*from:*	*departure_m*
León:	ciudad_origen	*León:*	*departure_city*
a:	m_destino	*to:*	*goal_m*
Córdoba:	ciudad_destino	*Córdoba:*	*city_goal*
para el tercer:	fecha_relativa_s	*for the third:*	*relative_date_d*
domingo:	dia_semana_s	*Sunday:*	*week_day_d*
de agosto:	mes_s	*of August:*	*month_d*
Cases			
(HORA-SALIDA)		*(DEPARTURE-TIME)*	
CIUDAD-ORIGEN:	León	*DEPARTURE-CITY:*	León
CIUDAD-DESTINO:	Córdoba	*CITY-GOAL:*	Córdoba
FECHA-SALIDA:	19-08-2002	*DATE:*	19-08-2002

Fig. 3. An example of the segmentation of an user turn. The Spanish original sentence and its English translation are given.

catenated in every possible order.[4] With this strategy we try to achieve an automatic division of a multiclass sentence in each of its constituting parts, each belonging to a different frame type. This is a natural approach for many turns (e.g. "*Yes, what type of train is it?*") but it is not so clear if this approach will be adequate for other turns, where such a clear division between the frames can not be found (e.g. "*I want information about timetable and prices for traveling form Barcelona to Vigo.*").

4.1. Smoothing

We are working with a closed vocabulary task, as the input from this phase is the output of the speech recognizes, which has a limited (and known) vocabulary. However, we can not assure that every word of the vocabulary will be seen in the training process, so, in order to handle these words in a proper way when they appear in the test phase, we need to apply smoothing techniques.

The emission probabilities of a word w in state s is given by[5]

$$\hat{p}_{sw} = \frac{N_{sw}}{\sum_{w'} N_{sw'}}\,, \tag{4}$$

where N_{sw} gives the frequency of word w appearing in the semantic unit (state) s. In this equation we can see clearly that the emission probability of a word not appearing in the training set will be set to 0. The first smoothing technique we have tried is known as Laplace smoothing and consists simply of adding a pseudocount ε to each word, thus equation (4) becomes

$$\hat{p}_{sw} = \frac{N_{sw} + \varepsilon}{\sum_{w'}(N_{sw'} + \varepsilon)}\,. \tag{5}$$

Normally $\varepsilon = 1$ but we consider a more general case where ε can be any non-negative real number.

The second smoothing technique we have applied is known as uniform backoff and consists of subtracting some probability mass from the seen events (words) and adding them to the unseen ones. The estimated probability becomes

$$p_{sw} = \begin{cases} \dfrac{N_{sw} - b}{\sum_{w'} N_{sw'}} & \text{if } N_{sw} > 0\,; \\ M \dfrac{1/|\Omega|}{\sum\limits_{w':N_{sw'}=0} 1/|\Omega|} & \text{if } N_{sw} = 0\,, \end{cases} \tag{6}$$

where $|\Omega|$ is the size of the vocabulary, b is an smoothing parameter ($b \geq 0$) and M is the gained probability mass given by

$$M = \frac{b \cdot |\{w' : N_{sw'} = 0\}|}{\sum_{w'} N_{sw'}}\,. \tag{7}$$

5. EXPERIMENTS

The smoothing technique and its parameters were determined by using the same validation set defined for the connectionist classification experiments (20% of the training set), but the training of the models for the test phase was carried out using the whole training corpus (see Table 2).

As a measure of the correctness of the understanding process we use two figures: the "Frame Error Rate" (FER)

[4]Normally, multiclass user turns are composed of only two or three frame classes.

[5]Equation (4) corresponds to the maximum likelihood estimator of a multinomial, where we must choose only one element of the original population.

Table 4. FER and CER of the filled cases with the different models. First, the results with transcribed data are shown and, secondly, with the recognized data.

Model	MLP err. rate	Test FER	Test CER	UC CER	MC CER
Global	–	28.73	20.97	23.01	19.11
Specific	0.00	19.78	13.42	13.16	13.72
Specific	11.19	25.37	16.99	17.39	16.71
Global	–	47.38	32.54	34.58	30.02
Specific	0.00	40.30	25.97	25.53	26.06
Specific	48.13	42.91	28.68	29.77	27.29

and the "Cases Error rate" (CER). For the FER measure, a frame is considered correct when the type of frame and its associated cases are exactly the same as the reference. The CER is defined as the word error rate between the output frame slots (type of frame and cases) and the correct ones.

A global model was also trained with all the user turns in order to compare its performance with the specific models. Several experiments were carried out, comparing the performance of the global and the specific models. In every case, the HMMs were trained using the correct transcription of the user utterance (transcribed data), not the output of the speech recognizer (recognized data). Table 4 shows that the use of specific models clearly outperforms the global one, both with transcribed and recognized data. The best results are obtained using the *correct* classification of each user turn. The use of the output of the classifier degrades the performance, as expected, but it is still better than the global model performance. It is worth noting that the understanding process is able to recover from some classification errors when using the recognized data.

6. CONCLUSIONS

We have shown that connectionist classification is a successful approach for classifying a user turn given in natural language into a specific class or classes of frames. It can also be noted that stochastic models are also a good approach for the understanding task.

It must be taken into account that in some frame classes very few training samples are available, so the models are underestimated. We hope to improve the performance of the system by a combination of specific and global models: If the user turn is classified with a high level of confidence, we could use the specific understanding model and if it is not, we choose the global one. Similarly we can also measure the confidence of the output of the specific HMMs.

We also expect to improve our system by retraining the MLPs with the recognized data in order to reduce the classification error. Lastly, more dialogues are being acquired for having more samples for a better estimation of the models.

7. REFERENCES

[1] G. Riccardi and A. L. Gorin, "Stochastic Language Adaptation over Time and State in Natural Spoken Dialogue System," *IEEE Trans. on Speech and Audio*, vol. 8, no. 1, pp. 1–7, 2000.

[2] C. Popovici et al., "Automatic Classification of Dialogue Contexts for Dialogue Predictions," in *Proc. ICSLP*, Sydney (Australia), 1998.

[3] L. Levin et al., "Domain Specific Speech Acts for Spoken Language Translation," in *4th SIGdial Workshop on Discourse and Dialogue*, Sapporo (Japan), 2003.

[4] T. Fukada et al., "Probabilistic dialogue act extraction for concept based multilingual translation systems," in *Proc. ICSLP*, Sydney (Australia), 1998.

[5] C. Martínez et al., "A Labelling Proposal to Annotate Dialogues," in *Proc. LREC*, Canarias (Spain), 2002.

[6] W. Minker, A. Waibel, and J. Mariani, *Stochastically-Based Semantic Analysis*, Kluwer Ac. Pub., 1999.

[7] E. Segarra et al., "Extracting Semantic Information Through Automatic Learning Techniques," *IJPRAI*, vol. 16, no. 3, pp. 301–307, 2002.

[8] D. Vilar, M. J. Castro, and E. Sanchis, "Connectionist classification and specific stochastic models in the understanding process of a dialogue system," in *Proc. Eurospeech*, Geneva (Switzerland), 2003.

[9] L. Lamel et al., "The LIMSI Arise system," *Speech Communication*, vol. 31, no. 4, pp. 339–354, 2000.

[10] R. Pieraccini, E. Levin, and W. Eckert, "AMICA: The AT&T Mixed Initiative Conversational Architecture," in *Proc. Eurospeech*, Rhodes (Greece), 1997.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *PDP: Computational models of cognition and perception, I*, pp. 319–362. MIT Press, 1986.

[12] C. M. Bishop, *Neural networks for pattern recognition*, Oxford University Press, 1995.

[13] A. Stolcke et al., "Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech," *Computational Linguistics*, vol. 26, no. 3, pp. 339–373, 2000.

[14] M. J. Castro, D. Vilar, and E. Sanchis, "Uniclass and Multiclass Connectionist Classification of Dialogue Acts," in *Proc. CIARP*. 2003, Submitted.

[15] A. Zell et al., *SNNS: Stuttgart Neural Network Simulator.*, University of Stuttgart, Germany, 1998.